Class Exercise - 3.2

Name: Dylan Mumm
Assignment ID: Class cancelled
Submission: Save this Word document with your answers as a PDF file and upload the PDF file to Canvas.

Case in Point 3.2: Parallel Services WBS

The project management team at Parallel Services is having a debate about how to define tasks in the work breakdown structure (WBS). Ann, the project manager, wants to break tasks down into the smallest possible units. For example, she objected to a broad task statement called *Develop training schedule*. Instead, she suggested three subtasks:

1. Determine availability of training room
2. Determine availability of attendees
3. Select specific dates and training times.

Karen, another project team member, disagrees. She feels that the broader task statement is better because it allows more flexibility and will produce the same result. Karen says that if you break tasks into pieces that are too small, you risk over-managing the work and spending more time on monitoring than actually performing the tasks.

1. As a member of the team, would you tend to agree more with Ann or Karen?
    - I would agree more with Ann.
2. What are the pros and cons of each approach?
    - Subtasks:
        - + Makes clear what specifically needs done
        - + Makes clear what tasks specifically depend on anothers completion
        - + Makes clear the estimated duration of each task
        - + More intuitively allows allocating specific subtasks to particular people
        - + More specific record keeping will more accurately represent what went wrong in case of delay
        - − Specificity will make noise register as more significant than it is
        - − This noise can lead to wasted labor hours (for example, if you allocate determine availability of training room to person 1, determine availability to person 2, and select specific dates and training times to person 1, if person 1 gets their first task done before 2 they will have to specifically wait on person 2's completion)
        - − Greater initial planning time
    - Broad task
        - + Hides noise
        - + Allows for more grass roots and dynamic time coordination between developers
        - + Less initial planning time
        - − Obfuscates what tasks needed
        - − Obfuscates dependencies
        - − Obfuscates predicted time for each (unspecified) subtask
        - − Worse record keeping